Remarks

I. Introduction

This Supplemental Response completes the Amendment that was filed together with the Request for Continued Examination (RCE) on October 31, 2007. In particular, the Amendment dated October 31, 2007, inadvertently omitted a reply to every ground of objection and rejection in the final Office action dated August 9, 2007. This Supplemental Response completes the Amendment of October 31, 2007, by providing the reply in accordance with 37 CFR § 1.111 as required under 37 CFR § 1.114. The Supplemental Response therefore completes the conditions required under 37 CFR § 1.114 for filing the RCE that was filed on October 31, 2007.

II. Claim rejections under 35 U.S.C. § 102

The Examiner has rejected claims 1-21 under 35 U.S.C. § 102(b) over Geigel (U.S. 2002/0122067).

A. Claim 1

1. The subject matter of claim 1

Independent claim 1 recites:

> 1. A method for arranging a set of objects within an area, comprising:
>
> initiating a first current binary tree comprising a leaf node;
>
> associating a first object selected from the set with the leaf node;
>
> establishing candidate binary trees, wherein each of the candidate binary trees comprises the current binary tree and a respective leaf node associated with another object selected from the set, and locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area;
>
> computing a respective score for each of the candidate binary trees

selecting one of the candidate binary trees as the current binary tree based on the computed scores;

repeating the establishing, the computing, and the selecting until the current binary tree includes all the objects in the set; and

after the repeating, arranging the objects within the area in accordance with the locations of the leaf nodes within the current binary tree.

2. Geigel does not disclose the establishing element of claim 1

As explained in the Amendment dated May 22, 2007, Geigel does not disclose establishing candidate binary trees in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1.

Geigel's page creator module assigns images in a collection to album pages based on a first genetic evolution algorithm. In this process, the page creator module evolves a genome that can be represented as a tree structure in which the root 148 represents the entire photo album, the intermediate nodes 150-156 represent individual pages of the album, the intermediate nodes 158-164 represent subgroups within the individual pages, and the leaf nodes 166 represent the images (see FIG. 8 and ¶ 89). However, the positions of the leaf nodes within the tree structure shown in FIG. 8 do not correspond to relative positions of the associated images within the area (i.e., album page) within which the images will be arranged. Instead, the positions of the leaf nodes 166 simply assign the images to respective pages and to respective subgroups within the pages.

Geigel's image placement module uses a second genetic evolution algorithm to generate genetic structures of page layouts for images that are assigned to a given page. In accordance with this algorithm, the image placement module determines the absolute positions of the images on each of the album pages by evolving a genome of the type shown in FIG. 17. This genome consists of a set of four positioning parameters (i.e., x-position, y-position, scaling, and rotation), which determine the absolute positions of the images to be placed on a given album page (see ¶ 126). However, neither the positioning parameters nor the genome of which they are a part constitute leaf nodes of a binary tree and, therefore, they cannot possibly have locations within a binary tree that correspond to relative positions of the associated objects within the area.

In response to this point, the Examiner has stated that (see page 6 of the final Office action):

> I.R Examiner does not agree, the term candidate binary tree is only merely describing is known in the art as a regularly binary tree, being that is the term candidate only yields to the known tree of insertion/ use of that tree, by definition of candidate. Also the leaf nodes of each binary tree correspond to relative positions as described in paragraphs 62, 69-73, 89, 112, 128 and 152.

This reply, however, does not address applicant's point that Geigel does not disclose establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. In particular, neither the discussion regarding the term "candidate" nor the Examiner's remark that "the leaf nodes of each binary tree correspond to relative positions" does not show that Geigel discloses establishing binary trees in which the locations of the leaf nodes within each of the candidate binary trees correspond to relative positions <u>of the associated objects within the area</u>. As explained in the following paragraphs, the references to paragraphs 62, 69-73, 89, 112, 128, and 152 do not make-up for the unresponsiveness of this reply.

Paragraph 62 reads as follows:

> [0062] Emphasis/Image Appeal--Images have an associated image appeal or emphasis value. This value is a measure of relative importance and is be used as a guide in determining the emphasis that an image will be given when placing it on an album page.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. Instead, this paragraph simply indicates that an image appeal or emphasis value is used as a guide in determining the emphasis that an image will be given when placing it on an album page.

Paragraph 69 reads as follows:

> [0069] 1) Coding--Genetic algorithms maintain populations of problem solutions. During implementation, these solutions are represented by some sort of data structure. The data structure used by a genetic algorithm is known as a genome. In the coding task, a data structure is chosen to represent the genome for the problem space and a mapping from the data structure fields to the problem domain is established Common genome data structures used in genetic algorithms includes lists, arrays, and trees.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. This paragraph describes characteristics of a genome data structure that is used in a genetic algorithm. This paragraph does not even hint that locations of the leaf nodes within the tree-type genome data structures correspond to relative positions of the associated objects within the area.

Paragraph 70 reads as follows:

> [0070] 2) Definition of Genetic Operators--New solutions are created via crossover and mutation of individuals from previous generations. Given a particular genome structure, the means for performing these operations must be defined. During crossover, one or more children solutions are derived from two or more parents. With mutation, new individuals are generated by mutation of a single solution. There are standard crossover and mutation operators available for genomes encoded using commonly used data structures like lists, arrays, and trees.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. This paragraph describes operations performed on a genome data structure used in a genetic algorithm. This paragraph does not even hint that these genetic operations involve establishing binary trees in which the locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area.

Paragraph 71 reads as follows:

> [0071] 3) Fitness--The most challenging and application specific task in applying genetic algorithms to a problem domain is in the definition of a fitness function. The fitness function is responsible for judging individual solutions and returning a score based on its evaluation. In essence, the fitness function defines the difference between a good solution to a problem and a bad one. Much care must be taken in defining the fitness function, as the genetic algorithm will converge on solutions deemed "fit" by this function.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. Instead, this paragraph describes a fitness function metric that is used to evaluate solutions determined by the genetic algorithm.

Paragraph 72 reads as follows:

> [0072] FIG. 3 illustrates a genetic crossover operation for trees 80 and 82. For the crossover operation, random nodes 88, 90 are selected for each parent and the sub-tree from these chosen nodes are swapped. This results in the `after crossover` trees 84 and 86, and the illustration demonstrates the change that occurs. Similarly, FIG. 4 illustrates a genetic mutation operation for tree 92. Nodes from the tree 96 and 98 are randomly chosen, then swapped. Tree 94 illustrates the change from tree 92 when this swap occurs.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. This paragraph describes genetic operations performed on a tree type genome data structure. Geigel does not even hint that these genetic operations involve establishing binary trees in which the locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area.

Paragraph 73 reads as follows:

> [0073] FIG. 5 and FIG. 6 illustrate similar genetic crossover and mutation operators, but in the case of an array type data structure. In FIG. 5, for a crossover, a position 101 within the array 100 is

> randomly selected and the array values 104 and 106 after the chosen position 101 are swapped between the two parents, resulting in crossover array 102. Similarly, in FIG. 6, two random array elements 112 and 114 are selected within a single parent 108 and their respective positions are swapped to produce resultant array 110.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. This paragraph describes genetic operations performed on an array type genome data structure. This paragraph does not even hint that these genetic operations involve establishing binary trees in which the locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area.

Paragraph 89 reads as follows:

> [0089] The genome for the page creator module makes use of a tree structure as illustrated in FIG. 8. The root of the tree, node 148, represents the entire photo album. Interior nodes, 150, 152, 154, 156, 158, 160, 162, and 164 represent a structure of hierarchical visual groupings of images, which, in turn, are represented by the leaf nodes 166. However, it is equally suitable to replace this tree based encoding with a simpler data structure based on arrays. The first layer of the tree 148 indicates the partitioning of the album into pages. Nodes below the first layer represent visual groupings within pages 150, 152, 154, 156. The notion of this hierarchical visual grouping is illustrated in FIG. 9, which shows the encoding of an album with two pages and a possible layout solution that maintains the visual grouping relationships. The album is represented by node 168. It comprises two pages 170 and a72. Page One has three images 180. Page Two 172 has three subgroups of images 174, 176, and 178. Each of these subgroups has two images. Subgroup 174 having images 182, subgroup 176 having images 184, and subgroup 187 having images 186.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited

in claim 1. The positions of the leaf nodes 166 within the tree structure shown in FIG. 8 do not correspond to relative positions of the associated images within the area (i.e., album page) within which the images will be arranged. Instead, the positions of the leaf nodes 166 only assign the images to respective pages and to respective subgroups within the pages.

Paragraph 112 reads as follows:

> [0112] To illustrate the relationship of these functions and clarify the process generally, what follows are several examples of using the Page Creator Module on a group of images. In each of the examples, the image set presented in FIG. 10 is used. While the images themselves are not present, the data is derived from actual user images and represents a typical user role of film consisting of 40 images taken from a variety of different events. To facilitate viewing the effects of the various evaluation criteria, the images have been labeled with a chronology index (C), giving the relative ordering of the picture with respect to the entire set. An assigned emphasis value (E) is also supplied with each image. Thus, referring to FIG. 10, groups of images from six events are illustrated. Event One 188 includes twelve imaged, chronologically C:0 through C:11. Event two 190 includes nine images, chronologically C:12 through C:20. Event three includes eight images, chronologically C:21 through C:28. Event four 194 includes six images, chronologically C:29 through C:34. Event five 196 includes four images, chronologically C:35 through C:38. Finally, event six 198 includes one image, chronologically C:39. Each image has a emphasis value which can be determined by reference to FIG. 10.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. This paragraph simply describes an exemplary set of images for manipulation by the page creation module in an illustrative embodiment. As explained above, the page creator module evolves a genome that can be represented as a tree structure in which the root 148 represents the entire photo album, the intermediate nodes 150-156 represent individual pages of the album, the intermediate nodes 158-164 represent subgroups within the individual pages, and the leaf nodes 166 represent the images (see FIG. 8 and ¶ 89). The positions of the leaf nodes 166 within the tree structure shown in FIG. 8 do not correspond to relative positions of the

associated images within the area (i.e., album page) within which the images will be arranged. Instead, the positions of the leaf nodes 166 only assign the images to respective pages and to respective subgroups within the pages.

Paragraph 128 reads as follows:

> [0128] The x and y positioning parameters give the placement of the center of a given image on the album page. In the genome, this is expressed relative to the total height and width of the page with the origin being the upper-left comer of the page. Appropriate calculations are made by the Image Placement Module to assure that an image placement calculated from given gene values will not result in any part of the image being placed off the boundaries of the album page. The mapping from the gene values for x and y positioning to actual x and y position on the page can thus be given by:

$$x = \frac{W_{image}}{2} + (gene_x \cdot (W_{page} - W_{image}))$$

$$y = \frac{H_{image}}{2} + (gene_y \cdot (H_{page} - H_{image}))$$

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. Indeed, this paragraph does not even hint that the positioning parameters are determined from or correspond to the locations of the leaf nodes within a binary tree.

Paragraph 152 reads as follows:

> [0152] The emphasis score measures the proportionality of the size of the images with respect to the emphasis values assigned to the images. The rationale behind this evaluation stems from the notion that images with large emphasis values should take up more space on the page. The comparison made during evaluation is relative to the sizes of all of the images. For each image, the size relative to the largest image in the group is calculated and expressed as a percentage. This percentage is then subtracted from the emphasis value assigned to the image. Then mean difference amongst all the images on the page is calculated and this average is

> subtracted from 1.0, assuring that an emphasis score of 1.0 indicates a strong positive correlation between image size and emphasis values.

This paragraph, however, does not disclose anything whatsoever about establishing binary trees (candidate or otherwise) in which the "locations of the leaf nodes within each of the candidate binary trees correspond to relative positions of the associated objects within the area," as recited in claim 1. Instead, this paragraph describes an emphasis score that is used to evaluate solutions determined by the genetic algorithm.

3. <u>Geigel does not disclose the repeating element of claim 1</u>

As explained in the Amendment dated May 22, 2007, Geigel also does not disclose the process of "repeating the establishing, the computing, and the selecting until the current binary tree includes all the objects in the set," as now recited in claim 1. In fact, Geigel does not appear to provide any details about how the initial solutions that are evolved by the page creator module and the image placement module are determined. In any event, Geigel does not even hint that these initial solutions are determined by iteratively (i) establishing candidate binary trees comprising the current binary tree and a respective leaf node associated with another object selected from the set, (ii) computing respective scores for the candidate binary trees, and (iii) selecting one of the candidate binary trees as the current binary tree based on the computed scores.

The Examiner's only reply to this point consists of the following statement (see page 7 of the final Office action):

> 3.A. Applicant argues on bottom of page 11 and top of page 12 the newly added claim language of claim 1.
>
> 3.R Examiner does not agree and notes attention to applicant to the new analysis of claim 1 above. Examiner still believes that the current condition of the claim language does not overcome the Geigel reference provided.

In the "new analysis of claim 1," the Examiner merely has stated that Geigel discloses the repeating element of claim 1 in paragraph 64, line 3, where "a plurality of images to be placed in

a file explains the repeated step nature of b-d repeated until done (see page 3 of the final Office action). Paragraph 64 reads as follows:

> The page layout system 124 performs two separate, yet equally important tasks. Page creation 126, given a set of images, the system distributes these images amongst a set of album pages, such that each image is assigned a page upon which the image will appear. And, image placement 132, once the images have been assigned to pages, each individual page is laid out by positioning the images assigned to it. Therefore, for each image, placement, rotation, and scaling of the image on the page are assigned.

This paragraph, however, does not disclose "repeating the establishing, the computing, and the selecting until the current binary tree includes all the objects in the set," as recited in claim 1. This paragraph does not even hint that any of the operations performed by the page creator module and the image placement module involve repeating the establishing, the computing, and the selecting until the current binary tree includes all the objects in the set. Indeed, this paragraph does not disclose anything whatsoever about how to create the tree type genome data structures that are disclosed in Geigel.

4. Conclusion

For at least the reasons explained above, the rejection of claim 1 under 35 U.S.C. § 102(b) over Geigel should be withdrawn.

B. Claims 2-7

Each of claims 2-7 incorporates the features of independent claim 1 and therefore is patentable over Geigel for at least the same reasons.

C. Claim 8

Independent claim 8 recites features that essentially track the pertinent features of independent claim 1 discussed above. Therefore, claim 8 is patentable over Geigel for at least the same reasons explained above in connection with claim 1.

D. Claims 9-14

Each of claims 9-14 incorporates the features of independent claim 8 and therefore is patentable over Geigel for at least the same reasons.

E. Claim 15

Independent claim 15 recites features that essentially track the pertinent features of independent claim 1 discussed above. Therefore, claim 15 is patentable over Geigel for at least the same reasons explained above in connection with claim 1.

F. Claims 16-21

Each of claims 16-21 incorporates the features of independent claim 15 and therefore is patentable over Geigel for at least the same reasons.

III. Conclusion

For the reasons explained above, all of the pending claims are now in condition for allowance and should be allowed.

Charge any excess fees or apply any credits to Deposit Account No. 08-2025.

Respectfully submitted,

Date: November 21, 2007

Edouard Garcia
Reg. No. 38,461
Telephone No.: (650) 289-0904

Please direct all correspondence to:
Hewlett-Packard Company
Intellectual Property Administration
Legal Department, M/S 35
P.O. Box 272400
Fort Collins, CO 80528-9599